The User Manual of
C. A. C. A. O.

Computer **A**ided **C**omposition of **A**tomic **O**rbitals

A Package of Programs for Molecular Orbital Analysis
[PC Beta-Version 5.0 , 1998]

CARLO MEALLI and DAVIDE M. PROSERPIO

with major contribution of
ANDREA IENCO

Original Reference: *Journal of Chemical Education* **1990**, *67,* 399-402

Symmetry routines written by Klaus Linn (1991)
Walsh Diagram Legend and other revisions: José A. Lopez (1992)
Consulence and contributions: Angelo Sironi (1992-4)
Free Format routines originally provided by Luis Farrugia (1994)
3D Surface data generation: Enrique Pérez-Carreño (1997)
Modified EHT (ASED, Calzaferri): Piero Macchi (1997)
Molecule Symmetrization Routines Provided kindly by Tullio Pilati and Alessandra Forni (1997) (program SIMMOL) *J. App. Cryst.* **1998**, in press.

The package is created and distributed from:

Istituto per lo Studio della Stereochimica ed
Energetica dei Composti di Coordinazione
(ISSECC-C.N.R.) Via J.Nardi 39 - 50132 Florence (Italy)
Tel. +39-55-2346653 (243990) - FAX: 2478366
e-mail= mealli@cacao.issecc.fi.cnr.it

Present address of D.M.P. :
Dipartimento di Chimica Strutturale e Streochimica Inorganica
Universita' di Milano, Via Venezian 21 - 20133 Milano, Italy.
Tel. +39-2-70635120, FAX 39-2-70635288
e-mail= davide@csmtbo.mi.cnr.it

For any problem, please report to the authors with a copy of the input file.
Your help and suggestions will be appreciated.

Whatsnew

The package mantains the basic structure with the programs EHC (for extended Hückel calculations) and CACAO (for graphic analysis). The programs, written for DOS, are now operated under the Windows95/98 systems through the usage of an handy graphic interface.

The basic idea behind CACAO is to run MO calculations and to examine the results visually and intuitively via the uasge of different graphs (including 3D drawings of the Molecular Orbitals). Thus the examination of long numerical outputs is not strictly necessary (although these are easily produced). In any event, the important numbers can be retrieved interactively during the visual analysis of the graphs!

Generation of input files is greatly simplified with **free-format** directives characterized by **keywords** and the data, to be provided by the user, are minimal.

The ehmo method is still one of the simplest ways to determine the symmetry properties of the Molecular Orbitals, full of chemical information. Accordingly, much care is taken to permit the construction of models with the highest possible symmetry. As a very helpful tool, any generated molecule (trial input file) appears in a **graphic window** (run EHC with the option MOL). The molecular animation and the easy analysis of the geometrical parameters allow the user to establish the correctness of the model. On-line help allows to understand easily the meaning of each single operation.

One strategy is to return immediately to the ascii editor and make the opportune modifications to the input file (*trial and error* proceeding). The alternative strategy is to access the graphic **Molecular Editor** which allows to delete or add atoms, replace substituents, etc. While editing the molecule, an algorithm allows to adapt the nearest symmetry pointgroup (**Simmol** routines) in order to exploit the ehmo method at the best.

In the input file, the whole molecule can be constructed by defining only its asymmetric portion and by specifying the **point-group symmetry** (see the directive SYMM). The atoms, generated with a minimum number of internal coordinates, are then expanded by the symmetry elements of the pointgroup (please remember that the main symmetry axis of the molecule must coincide with the cartesian Z axis!). To have a better picture of the structure and an hardcopy always handy, the program PLUTO (Cambridge 1988) can be launched from EHC or CACAO while the molecule appears in the screen (same orientation).

Also crystallographic coordinates can be used as input and, in particular the program can process a Cambridge Structural Database coordinate file. In this case the **Molecular Editor** is automatically accessed and the experimental molecule can be manipulated to obtain the best model based on it. Thus the orientation consistent with the highest possible symmetry is selected (**Simmol** routines), bulky substituents can be discarded and replaced with new atoms defined through internal coordinates. Mixed usage of cartesian and internal coordinates (ATOM + INT or INTZM directives) is allowed.

Added flexibility in the construction of the input file is provided by the **Constants** and **Free Variables** lines. The constants (CONST) allow to define only once some repetitive parameters. The Free Variables (FVn), necessary to construct Walsh Diagrams, are more flexible than before. The complete series of variables can be defined with the starting value and its increment or it can be the result of an **expression** which depends on other free variables and which is evaluated at the run time.

Some modified algorithms speed up the ehmo calculations and even very large molecules (depending on the amount of memory available, the present version can now process molecules with up to 400 atoms and 3500 orbitals) can be analyzed fast enough with pentium processors.

The second part of the package (program CACAO) permits the visual analysis of the ehmo results. The improved quality of the graphic window is evidentiated by menus which simplify the different operations and the switching among Walsh Diagram, Interaction Diagrams and 3D drawings of the MOs. Concerning the latter, the best view is now easily fixed by interactive molecular rotations.

Through the menu most structural information may be retrieved as well as all of the quantities generated by the ehmo calculation, including the Mulliken analysis. Important parts of the latter can be now analyzed *via* new graphic capabilities such as the plot of the Reduce Charge Matrix (**RCM**) and that of the Molecular Orbital Overlap Popolation (**MOOP**). The RCM diagram permits an overview of the percentage contribution of a group of atoms (one or more) to a number of MOs within a given energy range (additionally, the contribution of the remaining atoms is also plotted). The MOOP diagram allows to evaluate visually how much each MO contributes to the overall overlap population between two

selected atoms. Again more MOs can be selected within a given energy range. Incidentally, after a FMO calculation, a special MOOP diagram can be constructed which allows to evaluate how much each MO contributes to build up the total overlap population between the two fragments.

A new feature allows to construct **3D surfaces** of the Total energy as well as of any chosen MO. Two independent groups of variables, which map the *nxn* grid, can be defined with the simple logic of the Free Variables. When CACAO processes a file containing the data relative to the bidimensional grid, it produces on request suitable ascii datafiles to be exported to other packages (such as Matlab).

As in the previous CACAO versions, the results of many different calculations can be graphically analyzed in a single interactive session. Each single drawing appearing on the PC screen can be printed with a keystroke (laserjet or postscript printers) without exiting the program.

GETTING STARTED

WARNING: The programs require at least 486 Hardware. The programs have been fully tested with DOS versions 6.x and full screen DOS-shells of Windows 95/98/NT.

1) Install the package (see message.txt).
   The location of the executables is in the directory **MOAN** (acronym for MO-ANalysis). The input files and the results of ehmo calculations must be in subdirectories of MOAN. Another directory, PRINTGL (containing the printing routines), must be at the same level as MOAN. The subdirectory **FILES** (\MOAN\FILES) is provided with some test input files (extension IN).

2) The programs can be run by either a DOS shell or by clicking in the WINCACAO icon under Windows. In the latter case, the basic operations are menu driven (as an option your favorite ascii editor can be selected, default=notepad).

   From the DOSshell, the command file (**EH.BAT**) controls both the EHC and CACAO programs by concatenating the operations.

   From the box opened on clicking on WINCACAO, browse the directory moan\files and select the file N2.IN. Than click on the CALCULATE button to run EHC (with MOL set). At the end, click on the button CACAO to construct an Interaction Diagram for the nitrogen molecule. Alternatively, the File CO2.IN can be selected and used to construct a Walsh Diagram relative to the bending of carbon dioxide. In this way the user can familiarize with most of the options offered by the package.

3) Each drawing appearing on the screen can be **printed** on a **Laserjet** or on a **Postscript Printer** (a postscript file can be created as well). Strike either the **P** or the **S** option (for printing) that appears in the line at the bottom of each drawing. Printing is performed by the routines **PRINTGL** located in the directory PRINTGL (at the same level as MOAN). PRINTGL is shareware which can be redistributed in its original form (***please, contact the PRINTGL distributors to fulfill their requirements***). A command file PL.BAT (resident in MOAN) is the interface between CACAO and PRINTGL. PL is automatically activated during an interactive CACAO session, as indicated above. The commands in the PL.BAT file can be modified for usage with specific printers (follow the instructions in the PRINTGL package)
   The drawings produced by CACAO can be also printed or reused in other applications (wordprocessor, Coreldraw, etc.) after the end of the interactive session. In fact, the relative HPGL files (namely, CACAOx.HGL and/or MOANx.HGL or PLUTO.HGL) stay in the directory MOAN till the next CACAO session.

4) Complete the reading of this document and *learn to make new **INPUT** files* by using your favoured EDITOR. Store the new file in a new subdirectory of MOAN (*e.g.* MYDIR) with the extension IN (e.g. NAME.IN). Then try a calculation.
After the ehmo calculation has been completed, there are in MYDIR two new files NAME.COR and NAME.BIN, respectively. The former contains the generated cartesian coordinates and the latter all the numerical information needed to run CACAO.

.

MAIN FEATURES OF THE PACKAGE

There are two basic programs:

1) **Program EHC** (Extended Hückel Calculation):

This is a major revision of the original program SIMCON (Cornell University) with weighted Wolfsberg-Helmholtz formula see: R. Hoffmann, *J. Chem. Phys.*, **1963**, *(39)*, 1397 R. Hoffmann, W. N. Lipscomb *J. Chem. Phys.*, **1962**, *(36)*, 2179, 3489, J.H. Ammeter, H.-B. Burgi, J.C. Thibeault and R. Hoffmann *J. Am. Chem. Soc.*, **1978**, *(100)*, 3686

Initially, a calculation of Cartesian coordinates from **internal coordinates** is performed (optionally **crystal coordinates** can be used by providing also the CELL parameters. Input cartesian coordinates are also treated as crystal ones with cell: A=B=C=1.0, alpha=beta=gamma=90). Many possibilities are offered to manipolate the model, either interactively or by modifying the input file at will. When the model is satisfactory, the actual MO calculation can be performed. The program outputs the information typical of MO calculations (energy levels, Wavefunction coefficients, Mulliken population analysis, etc.). The length of the output can be limited by using specific keywords (EL, WF, etc., see below a complete list).
The program automatically determines the **molecular symmetry** (pointgroup) and each MO level is assigned the proper symmetry class. In a single run it is possible to repeat the MO calculation at different molecular geometries (steps) by specifying in the INPUT file one or more geometrical parameters to be varied (**free variables**). In this manner, the program builds up the necessary information to make a **WALSH diagram**, later drawn by CACAO. Otherwise, it is possible to specify in the INPUT file two fragments, in which the molecule is ideally separated, so that the program can perform a **Fragment Orbital Analysis** (**FMO**). This means that separate MO calculations are performed for the molecule itself and for the single fragments. Eventually, information is obtained about the contribution of the fragment molecular orbitals (FMO basis set) to the main MOs, themselves. The information can be used later by CACAO to construct an **Interaction diagram** between the sets of FMOs. This type of analysis is very powerful to focus on the formation of specific chemical bond(s) within the molecule. Notice that, although the EHC allows the definition of more than two fragments and it calculates the composition of the MOs in terms of these multiple basis sets, the graphic routines of CACAO work only when two fragments are selected.

2) **Program CACAO** (Computer Aided Composition of Atomic Orbitals)

The program is interactive and reads the information in the files NAME.COR (cartesian coordinates, ascii) and NAME.BIN (binary) created by previous EHC runs. Multiple Dataset can be processed at one time.

The program can produce the following graphics:
a) Walsh diagrams or Interaction diagrams.
b) three-dimensional drawings of several MOs for one molecular geometry.
c) One MO at different geometries (i.e. at different values of the reaction coordinate in a Walsh Diagram)
d) One MO with its FMO components (if the option for Fragment Orbital Analysis (FMO) is activated.

If two sets of independent variables (FVn and Fv-m) have been selected in the previous EHC calculation, CACAO prepares an ascii file (NAME.ENE) which contains data for plotting a surface of the total energies. This file can be processed by typical commercial programs such as MATLAB (a command file MATLAB.M can be found in the directory Moan). Also in this case, the user may select one row or one column or one diagonal of the square surface of dimension IPOINTxIPOINT in order to analyze the data in the usual manner (Walsh Diagrams, 3D MO drawings etc.).

Hints on the preparation of the data for MATLAB and its usage:
Generate the grid of points by properly selecting the free variable Fvn and Fv-m and by running EHC. At the end, run Cacao. The graphic capabilities of the latter are not immediately available as usual. Thus, you are asked whether you want to generate a surface (of total energy or of the energies of a given MO) or to select any row, column or diagonal of the bidimensional grid in order to visualize a monodimensional dataset, as usual. In this latter case, you will be presented the list of available datasets which contains one or more new files corresponding to the selected colums, rows or diagonal. Select any of them by number and proceed as usual.

If a grid has been chosen, a new file is generated with extension “ene”. Also, the file name terminates with the characters “_s”, “_h” or “_l” for total energy, homo or lumo, respectively. Otherwise with the number of the selected MO.

At this point you can run MATLAB. From the menu open the file Matlab.M in the directory moan and change opportunely the name of the file xxxx.ene and save it. Then select from the menu the option for running the matlab.m command file. The surface will appear. You can modify the view point with the command: view(az, el), where az=azimuth and el=elevation.

In addition, CACAO allows a real-time visualization of the numerical quantities associated the **Mulliken population analysys** (composition of the MOs in terms of Atomic or fragment Orbitals, energies of the levels, overlap populations, charges etc.)

The program allows visualization of the graphics of type *a, b*, *c* and *d* and the numerical information associated with them. Easily, the user can switch back and forth among different datasets. During a single CACAO session, diagrams

and MO drawings can be generated for all of the structures with available files of the type *.COR and *.BIN. The results of many different calculations and their graphic representations can be then compared in real time. Also, hard copies of the graphs displayed on the screen can be produced at any moment (laser or inkjet color printers, by using the shareware package PRINTGL).

EHC (detailed technical information)

The following limits are presently imposed:

a) Max. number of atoms = 350
b) Max. number of orbitals = 3500
c) Max. number of steps in the Walsh diagram=20.
d) Max. number of free variables=50

The availability of physical memory in your PC is also a constraint, not as crytical as before as the programs are not restricted anymore by the 640K limit typical of DOS.

The ehmo calculation uses *cartesian coordinates*. The latter can be either given as input, or can be generated by the program starting from *crystallographic or internal coordinates*. The *internal coordinates* are most useful to reconstruct highly symmetrical molecules and are very helpful in fixing stereochemical rearrangements or reaction pathways. Their definition is recommended.
The internal coordinates can be based on either a *simplified* **Z-MATRIX** or a *complete* **Z-MATRIX**.
Essentially, each new atom is being defined by three preexisting atoms or points (*dummy atoms*). Figure 1 helps to understand the disposition of the four points necessary for each new atom definition: In the INPUT file, the latter points can be either fully explicated (complete Z-Matrix, INTZM card) or only partially explicated (simplified Z-Matrix, INT card).

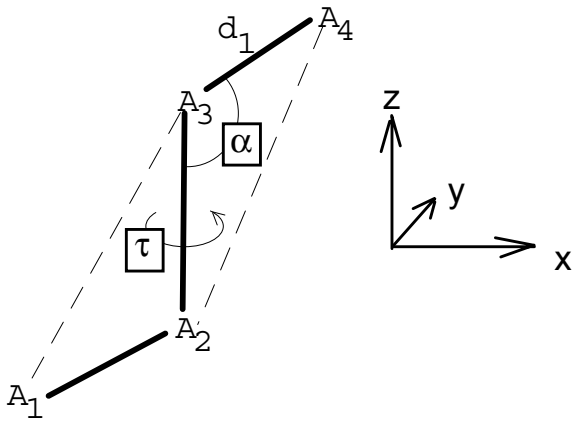


Figure 1

The atom with absolute number 4 (A_4) is being currently defined. This forms a vector A_3-A_4 of the given length (d_1 = first internal coordinate) with the preceeding atom number 3 (A_3). The angle at A_3 (= A_2-A_3-A_4) represents the second internal coordinate (alpha angle). Obviously, it is necessary to refer to the atom number 2 (A_2) from which A_3 was generated.
Finally, the third internal coordinate (torsion angle, tau) is given by the torsion angle defined by four atoms A_1-A_2-A_3-A_4 (A_1 generator of A_2). This is also the dihedral angle defined by two planes, namely:

i) the plane A_2-A_3-A_4 formed by the vector being defined (A_3-A_4) and the immediately preceeding one (A_2-A_3).

ii) the plane formed by the vectors (A_2-A_3) and (A_1-A_2), whose atoms have been previously defined.

In the simplified Z-Matrix (INT directive, shown below), it is assumed that the order of the four atoms is sequential. In other words, the pathway that allows to define A_4 is fixed and it goes through the previous atoms A_1, A_2 and A_3.

INT	SYMB	AT(N-1)	AT(N)	DIST	ALPHA	TAU
	A_4	**3**	**4**	**d1**	**α**	**τ**

By choosing a full Z-Matrix, the sequentiality of the four atoms is not required. Thus, one needs to specify each time (in a INTZM directive, shown below) the numbers of any four atoms (A_1, A_2, A_3 and A_4), with the first three however defined.

INTZM	AT(N)	SYMB	AT(N-1)	DIST	AT(N-2)	ALPHA	AT(N-3)	TAU
	4	**A_4**	**3**	**d1**	**2**	**α**	**1**	**τ**

Notice that the orders of the parameters in the INTZM card is consistent with that of other quantomechanical programs. In preparing the INPUT for EHC the INT or INTZM directives may be mixed at will.

The sign of torsion angle tau depends on the movement needed to bring the vector A_3-A_4 into the plane A_1-A_2-A_3. The sign is positive if this movement involves a righthand screw motion, and negative if this movement involves a left hand motion. Some confusion arises, when one has to decide whether tau has a value between 0° and 90° or between 90° and 180°. As a *rule of the thumb*, consider the vector A_2-A_3 as the binding axis of a book. If the book appears open as in the Figure 1 (the vectors A_1-A_2 and A_3-A_4 lie on opposite sides with respect to A_2-A_3) then tau belongs to the second quadrant. Conversely, if the book is closing (A_1-A_2 and A_3-A_4 are *cis*-oriented), tau is in the first quadrant.
Also to become familiar with the torsion angle definitions, the new user is warmly recommended to practice by making on the paper stereochemical sketches of different molecules. Assign progressive numbers to the atoms and determine all of the internal coordinates (see one example, below). It is easy to draw with CACAO the generated structure (after interrupting EHC) and verify the correctness of the assignments, hence go back to make the appropriate corrections.
To set up any reference point (*e.g.* the centroid of a ring) the user can define *dummy atoms* (symbol DU and optionally a negative number). Dummy atoms are automatically removed from the ehmo calculations. The user can only *point once* to the same atom (only one definition) but he can point, as often as needed, from an existing atom to other ones. This will be clearer by following the next example relative to the ideal complex PtL_3(Etylene).

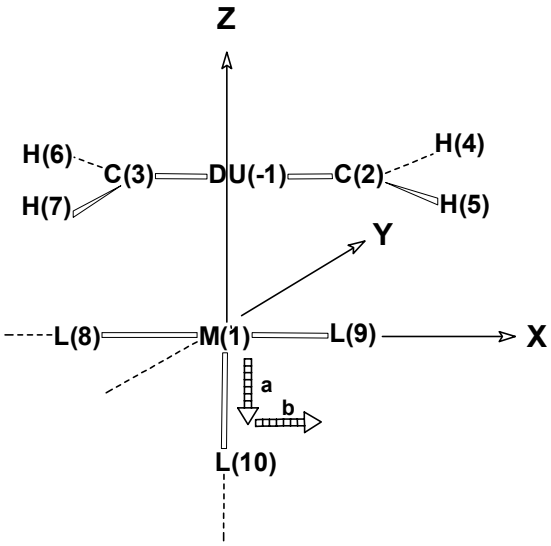


Figure 2

The first atom Pt(1) in the input file defines the origin, and it is fixed by providing its cartesian *x,y* and *z* coordinates (line 1, in the box below). Notice that the origin can be a *dummy atom* in some cases. The first two atoms departing from the origin require the usage of two **predefined vectors** to make explicit the alpha and tau angles (lines 2-3). These directions are the cartesian -Z (pointing toward negative Z values, arrow **a** in the Figure2) and the +X axes (arrow **b**), in the order. For example if we choose an angle alpha of ±180° the dummy atom DU(-1) is on the Z-axis in positive direction and the torsion angle is undefined (0°). Now we define C(2) by fixing the angle M-DU-C at 90°. This is not sufficient as also a torsion angle (tau) is needed. C(2) lies in the X-Z plane for both the tau values of 0° and 180°. If C(2) has to eclipse the +X direction, the the torsion angle is 0° (closed book). Obviously, the second carbon atom C(3), on the opposite side with respect to C(2) has tau=180° (open book). The hydrogen atoms linked to C(2) are defined by the three previously defined atoms, *i.e.* C(2), DU and M (in descending order). The user does not have to waist much time in defininig the tau angles for H(4) and H(5) as, if one is 90° the other must be -90° (or 270°).

```
ORIG 0 0 0 Pt                          ORIGIN M(1)
INT  1  -1   DU  2.0   180  0.         dummy atom
INT -1   2    C   0.66   90.  0.       C(2)
INT  -1  3    C   0.66   90 180.       C(3)
INT   2  4   H   1.0,120.,270.         H(4)
INT  2  5    H   1.0,120.,90.    H(5)
INT  3  6    H   1.0,120.,90.    H(6)
INT  3  7    H   1.0,120.,270.         H(7)
INT  1  8    L   1.7,90.,180.           L(8)
INT  1  9    L   1.7,90.,0.             L(9)
INT  1  10   L   1.7,0.,0.              L(10)
```

More examples (at the end of this document) will help to clarify the strategy followed, case by case.

By specifying the internal coordinates, the user determines the orientation of the molecule with respect to the cartesian basis set (X, Y, Z). **The main symmetry axis (twofold, threefold, etc.) must coincide with the cartesian Z-axis.**
Specific routines in EHC allow the determination of the **symmetry pointgroup** from the analysis of the atomic coordinates. After the calculation of the MOs and their energies, each MO is assigned the appropriate **symmetry label**. Failures are still possible for a restricted number of pointgroups. For example the program does not recognize the highest symmetries, such as I_h or T_h , while in T_d and D_{5d} pointgroups not all of the classes are properly assigned to the levels. Moreover, the symmetry is properly recognized only if the molecule is centered on the origin of the cartesian system with a canonical orientation of the axes (the same as that in the Character Tables for Symmetry Groups in "Chemical Applications of Group Theory" by F.A. Cotton).
Warning: in the FMO calculations, the fragments are not searched for the highest symmetry (no molecular riorentation is made). Consequently, the fragment may appear to possess lower symmetry than the actual one. A warning message is issued by EHC.

As mentioned, crystallographic or cartesian coordinates can be used in the input file, but the usage of the internal coordinates is still warmly recommended. One of the reasons is that only by using internal coordinates it is possible to construct Walsh Diagrams, hence to monitor the evolution of the MOs for a certain geometric deformation or reaction coordinate.
In order to generate one of these pathways, **free variables** (FVn) can be defined (distances, angles, torsion angles as well as rotation and translation of **rigid groups** in the molecule, see below) and later referenced to in the internal coordinates (INT) or group (GR) lines.
Some of the most common molecular fragments or groups can be internally generated by the program with only one instruction line as they were single atoms (e.g. **predetermined groups** such as phosphine, amine, cyclopentadienyl, carbon monoxide, phenyl, etc. are introduced as they were single atoms with the special symbols FO, AM, CP, CM, PH, etc.)
In order to have EHC performing a FMO analysis, the **keyword FMO** must be specified in the input file, as a single line directive following all of the atomic definitions (internal, cartesian or crystallographic coordinates). Additional information about the composition of the fragments must follows in the subsequent lines. It is also possible to have an **FMO calculation by using a file already set for a Walsh diagram** (i.e. containing free variables). Indeed, if the FMO instructions are present in such a file, they are taken into consideration only if the keyword FMO is followed by an integer specifying the step number of the corresponding Walsh Diagram (normally set to zero). The FMO calculation supersedes the creation of the Walsh diagram, in this case.

For a WALSH diagram, be sure that the **symmetry remains constant at all steps**. If, at a given step, the symmetry becomes higher, keep it lower by introducing small deformations for the appropriate parameters (usually a bond length variation of 0.01Å or angular deviations of 0.1° for angle or torsion angles is sufficient).

The program is organized to save the largest amount of **printed output**. Notice that the elements of the **Charge Matrix** are printed with the same sign as the corresponding elements of the **Wavefunction Matrix**, so that it is usually not necessary to print both matrices in order to determine the composition and the phase of a given MO. Also, the elements of the Charge matrix are normalized to 1 rather than 2 (electrons), so that the actual values can be derived as: *Charge matrix element=ABS(printed value)/500*. These magnitudes can be taken as the **percent contribution** of a basis set orbital (atomic or fragment) to a given MO.
The possibility of retrieving the most significant numerical quantities at any time on the screen (by running CACAO) greatly reduces the need for printing the output, unless some specific magnitudes need to be analyzed (e.g. Overlap integrals, Hückel matrix, etc.)

CACAO (technical information)

More than one set of calculations can be analyzed during an interactive session of CACAO.
At the start, all of the **available datasets** in the working directory can be listed and selected by their corresponding number. Later, the list can be reexamined by quitting the present selection (key Q in the menu line) and subsequently by asking for a new Dataset (key D). Once the graphics have been constructed, it is possible to jump back and forth between them, although relative to different models and calculations. In this manner, the features are compared visually.
In general, the user is driven by the **questions** appearing in the menus, which are in the simplest form, *e.g.* Yes or No (Y/N), a numeric code (1, 2, etc.), or a key (Z, G, etc.). Most of the times, it is sufficient to type return (CR) to select the **default** option (**DEF**).
Basically, one can construct four types of different diagrams (Interaction, Walsh, Reduce Charge, MOOP) or 3D drawings of the MOs and switch them, at will. For example, the user selects one MO by looking at an **Interaction diagram**, then he can examine the level and also its FMO components. Given a **Walsh Diagram,** the user may follow the evolution of the MO along the pathway and try to understand from the variation of the shape (bonding/antibonding character) why the level descends or rises in energy along the pathway. The **Reduced Charge Matrix** (**RCM**) diagram permits to overview the percentage contribution of two different groups of atoms to all of the MOs within a given energy range (additionally, the contribution of the remaining atoms is plotted). The **MOOP diagram** allows to evaluate visually how much each MO contributes to the overall overlap population between two selected atoms.
To have more quantitative information handy, the user can ask interactively for the actual numbers generated by the **Mulliken population analysis** (option 3 of the Menu). At this point, the composition of a given level is obtainable by typing the positive number of the MO in question. The latter magnitude (%) is accompanied by a sign (+/-) which specifies the **phase** with which a certain atomic orbital contributes to the given MO. Another possibility is to **check all the MOs containing a particular atomic orbital** (*e.g.*, enter: "7,px" in order to see how the px orbital of the atom number 7 is distributed among MOs).
If you are dealing with a dataset containg FMOs (Interaction Diagram), a negative number, given at the MO request, identifies the FMO characterized by the corresponding absolute number. The FMO composition is hence presented. If one enters two FMO numbers (both with negative signs and separated by a comma), the program identifies the MOs which receive contributions from both of them. This is a quick way to individuate bonding and antibonding MOs. Since the pairs of the FMOs which are involved in significant direct interactions are shown at the right side of the Interaction Diagram itself, CACAO does greatly simplify the exploration of the chemical bonding in the Molecule.
Also, the atomic coordinates, the geometric information, the **reduced overlap population** (**ROP**) between two atoms, etc. can be retrieved under the Menu 3.
If the current dataset is relative to a Walsh Diagram, one can **select the step of interest** (by entering the corresponding number as a negative value) and make all of the possible inquires about it. Moreover useful **tables relative to the variations of a given magnitude for all of the steps** (*e.g.*, ROP or atomic

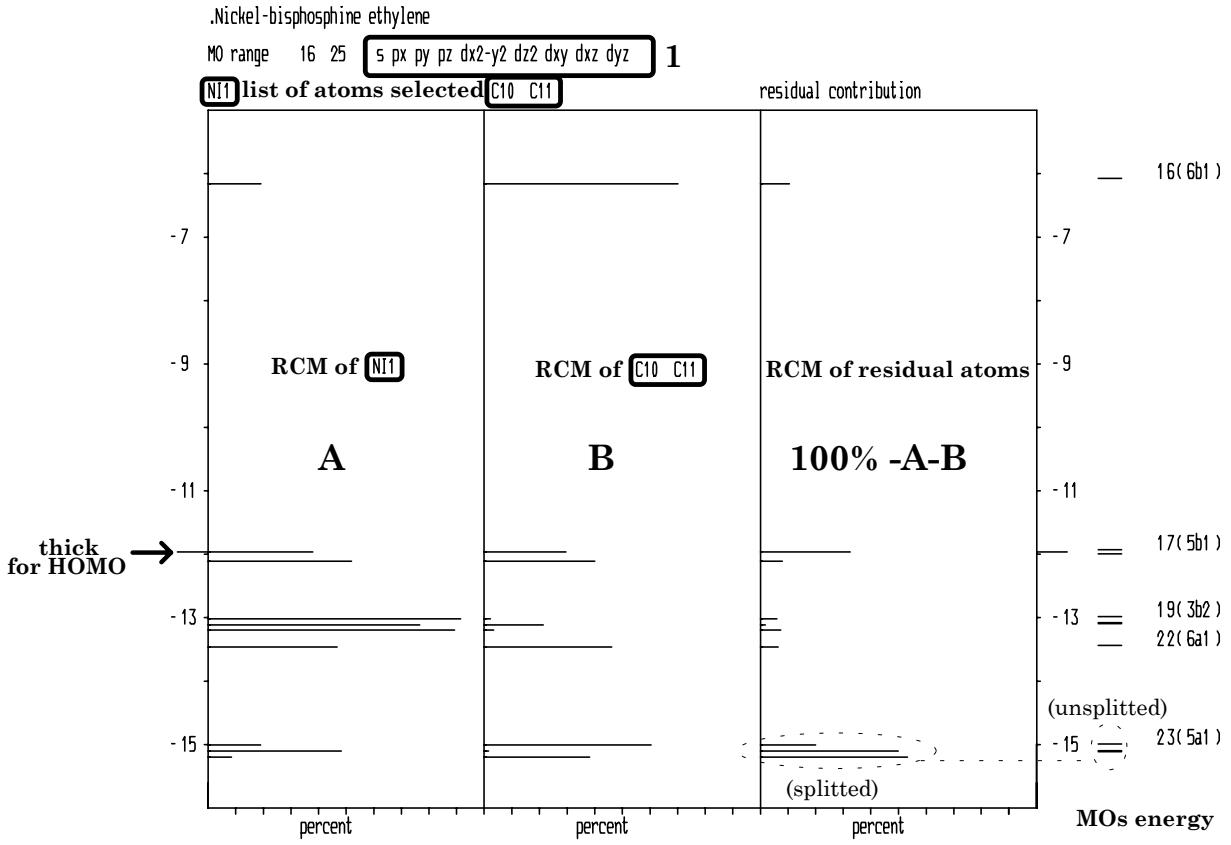
charge) are automatically generated by entering one, two or three atom numbers, separated by commas. At any moment, the user can switch back to the drawings to investigate additional features, hence he can move again to the retrieval of associated numeric quantities. The above procedure can be repeated for different datasets, as well as it is possible to switch between many Interaction, Walsh and MO drawings. Recall that the files opened at any moment are those relative to the picture on the screen. This means that the subsequent new pictures or numerical quantities to be recalled refer to that particular dataset. Any **previously drawn picture** can be recalled by its number made negative (e.g. -3 recalls the third drawing). If the picture in question belongs to a different dataset, the corresponding files are reopened so it is possible to do more work with it.

When building an Interaction or Walsh Diagram, the user can select the **width of the energy window**. This can be varied at will in order to focous on a selected group of MOs (usually the Frontier ones) that carry relevant chemical information. Essentially, the user supplies the length of the Y axis (in eV). Optionally, he can also **select the levels** to be plotted, based on their symmetry class or more simply on their progressive numbering.
In the Interaction diagrams a **threshold for the minimum FMO contribution** to a given MO is provided. In this manner, the connection lines between MOs and FMOs can be limited in number. Also, the connecting lines appear with different colours, each colour corresponding to a different **percent** (**%**) **group** (as an indication, the legend at the top of the diagram prompts that a green line corresponds to a maximum of 50% contribution of the FMO to the connected MO). In this manner, one gains a quick idea of the composition of a certain MO withouth looking at the actual numbers. The latter can be anyway retrieved through the Mulliken population analysis (option 3 of the menu). The **legend, at the right side of the interaction diagram** reports the number and the proper symmetry labels of the each MO and FMO. A little segment shows their actual energy position. Notice that in the main diagram itself, the levels which are degenarate or almost degenerate, are artificially shifted downward to avoid overlap. Optionally, the downward shift of the levels is prevented, so that a more realistic picture of the level distribution is shown, although many details are lost.
In the Walsh diagram, an option allows to **revert the original order of the steps**. For example, if the free variable for the generation of the diagram is the progressive elongation of a certain bond (from the left to the right), it may be possible to interpret the diagram in terms of the formation of the same bond from two separate fragments. The inversion of the steps is trivial but it result conceptually cleaner. The **total energy** is optionally plotted in the Walsh diagram as a **dashed red line**. One can choose whether to report the latter on an **absolute scale (1div.=1eV)** or on a **relative scale**. In the latter case (default), the intervals on the ordinate (which are meant to represent the energy scale of the single MOs in eV units) are used as a reference for the total energy. However, each divison now corresponds to an amount of energy that is reported in the **legend at the bottom**, together with the minimum and maximum total energies.
In the Walsh Diagrams, the filled levels are green, while the empty ones are yellow. Singly populated levels have also a **different color**. As new nice

feature, when the electron population changes along the pathway, so it does the colour of the corresponding MO. This may happen when two frontier levels become almost degenerate so a change of ground state is also assumed. However, an option is provided for keeping the latter constant (see below, directive 2). In very complicate cases, when the population of the frontier levels changes more than two times along the pathway, an option is offered to plot all the levels as they were apparently empty (yellow lines).
Up to 12 three-dimensional drawings of the MOs can be shown at one time on the screen (for better resolution it is better not to exceed the number of **six drawings**). If dealing with a Walsh diagrams, one can visualize **more MOs at a given step or the same MO through different steps**. If dealing with an Interaction diagram, one can visualize more MOs or **one MO with its FMOs components**. Also one can ask to draw only a certain number of FMOs.
The MO drawings are automatically scaled on the screen, whereas the scale for the **hard copy** (HPGL file) is defined by the user immediatly before the calculation starts. The options are: 1=**1cm/Amg**, 2= automatic rescaling to fit one printed page (default), 3= allows the introduction of a used defined scale. The usage of the options 1 or 3 may cause the problem that the printing of all of the drawings on the screen may require more than one page. On the other hand, automatic rescaling over one page produces very large drawings even if the molecule is small (*e.g.* N2).
To mantain as much as possible the identity of the Atomic Orbitals in a given MO (as in the **Hoffmann type sketches**) the AOs are artificially contracted by a factor of 1.5 (default). Usually, the **contraction** avoids the overlap between in-phase atomic orbitals of adjacent atoms. With very diffuse orbitals, a larger contraction coefficient should be selected to reach this goal. Otherwise, to visualize the real MOs, and not their artifacts, the user restores the **contraction coefficient** to its natural value of 1.
To improve the quality of a drawing (at expenses of computing time), adjust the finess of the grid. The **grid finess code** ranges between 1 and 9, the latter corresponding to the coarsest grid (DEF code=6). The code is accompanied by a key-letter which specifies the **drawing mode** (i.e. the spatial relation between the viewer and the molecule as well as the density of the countour lines). The valid key-letters are suggested by the program, for example the letter X allows to draw only the **MO envelopes** and it is the fastest way to obtain a draft of the MO drawing.
In order to speed up the drawing of the given MO, select only the atoms whose atomic orbital contributions has to be highlighted. Thus, the **Atom Selection Code** 0 is assigned to a given atom only if its atomic orbitals provide a significant contribution to the MO (coefficients >> 0) or if such contribution is to be evidentiated. The codes 1 or 2 neglect any possible contribution of the given atom in constructing the MO drawing (for example, the atom is not very significant for the chemical analysis as it may be the case for the inner parts of a bulky ligand in a transition metal complex). Notice that the atoms whose orbital contribution is neglected can be optionally represented in the structural skeleton (Code 2 = Show only the atom). Finally, the Selection Code 1 cancels from the drawing any presence of the atom in question. To speed up the procedure for the **assignment of the codes**, hence the input operations, an unique code can be applied to a certain group of atoms. The **atom range** runs from the current atom to the one higher atom, specified by the user. The range

is homogeneus, thus it includes only the the atoms in the lists which are of same type as the current atom. When defining the range, the letter A (=all), specified in place of the upper atom number, means that the given Selection Code (0/1/2) applies to all of the atoms of that type.
If some of the atoms are selected (code 0) while some of those related by a symmetry element(s) (planes or center) are discarded (codes 1 or 2) you may have a wrong MO picture. Redraw the picture after a **Symmetry Usage Redefinition** (the option becomes available in the very **last question** before the computations for drawing are started). Discard the use of any symmetry available to the program by entering the string XXXX after you have asked for Symmetry Redefinition with the letter S (normally at this point the letter P or CR is striken for continuing the job). In general, discard the usage of the available symmetry information when unusual features in the drawing are observed. It cannot be excluded that EHC has assigned a wrong symmetry to the given MO. Report the case to the authors, by supplying the INPUT file.

A graphic analysis of the Mulliken Population Analysis can be carried out by constructing two different types of diagrams, namely those of the Reduced Charge Matrix (**RCM)** and of the Molecular Orbital Overlap Population (**MOOP)**. This can be done by slecting the keys X (RCM) or (MOOP) which appear in the menu at the bottom of Interaction or Walsh Diagrams.
The **RCM** diagram displays the values of the Reduced Charge Matrix of selected atoms to each MOs within a range of energy, expressed in percent. The **RCM** Diagram is divided in three sections: in the left column shows the percent contributions (%A) of the first group of atoms to the MOs, in the central the contributions (%B) of the second group, and in the right column the contributions of the remaining atoms (that is equivalent to 100%-%A-%B). One line is drawn for each MOs, also for degenerate (*e.g.* 3 parallel lines for “t” levels), artificially splitted to show all of them. On the right the real position of the MOs is reported with labels (only for some of them, avoiding overlapping labels).
Entering the X=RedChg option the user has two cases depending if coming from Walsh or FMO diagrams.
In the case of FMO analysis it is possible to display a special RCM diagram with column A and B defined as the RCM contribution of the two fragments. In this case the third column is empty, because column A and B selected already all the atoms.
For a Walsh case the user must provide the step to be analyze.
After that, the user is asked to input the number corresponding to the atoms contributing to the two columns.
If only one atom is selected (for column A and/or B), the RCM line is multicolored to shows the single AOs contributions to each MOs, the colour code for each AO is on the plot (**1** in figure below)
For people familiar with solid state calculation, the RCM diagram is parallel to the projected Density Of State (DOS).


.Nickel-bisphosphine ethylene
MO range 16 25
s px py pz dx2-y2 dz2 dxy dxz dyz
1
NI1 list of atoms selected C10 C11
residual contribution
RCM of NI1
RCM of C10 C11
RCM of residual atoms
A
B
100% -A-B
thick for HOMO
-7
-9
-11
-13
-15
16(6b1)
17(5b1)
19(3b2)
22(6a1)
(unsplitted)
23(5a1)
(splitted)
percent
percent
percent
MOs energy

The **MOOP** diagram displays the overlap population contribution to selected bonds/interactions for each MOs, within a range of energy, together with the integral value of the MOOP.
First the user select the interactions to be analyzed, *i.e.* the one or more bond to be averaged in the MOOP plot (bonds equivalent for symmetry should be averaged). Any couple of atoms could be selected, even at non-bonding distance, if interested in particular weak interaction.
The diagram reports a (partial) list of the selected bonds (**1**), the number of averaged bonds (**2**), the (averaged) value of the Reduced Overlap Population for the same bond(s), for the given occupation (**3**). The ROP value is also proportional to the lenght of the integral at the HOMO (see figure). The diagram is divided in two parts. On the left, the values of MOOP are displayed with the scale at the bottom, starting from the central “0” line. On the left of the line are the (red) antibonding contribution, and on the right the (green) bonding ones, to each MOs.
The right part report the integral value of the MOOP(IMOOP), defined:

$$\mathrm{IMOOP[MO(n)]} = \mathrm{MOOP[MO(n)]} + \mathrm{IMOOP[MO(n-1)]}$$

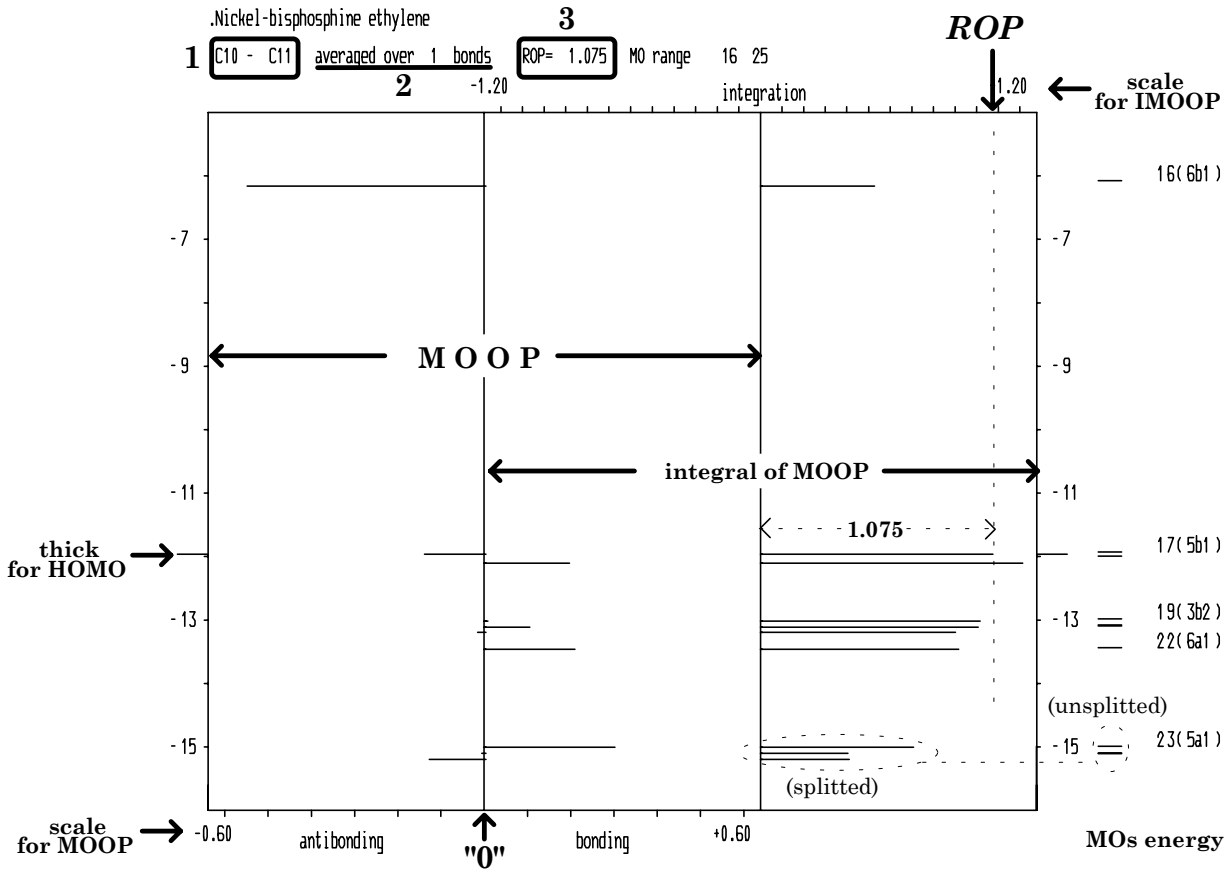
or, in a more general way (with *LOMO* = Lowest Occupied MO) :

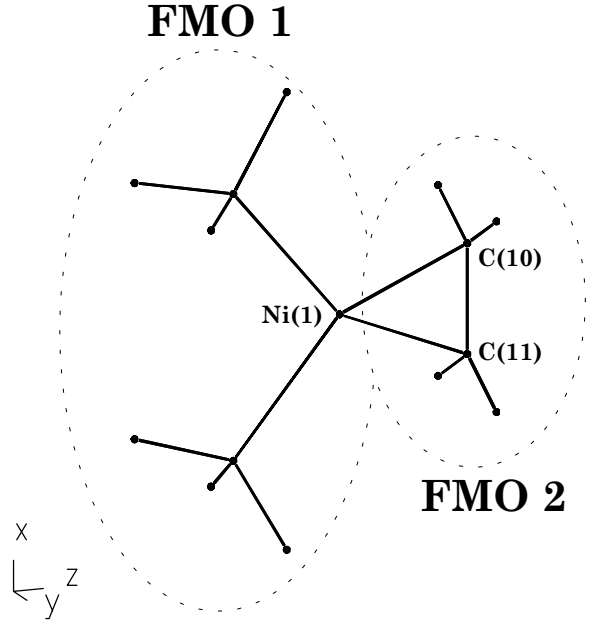
$$\mathrm{IMOOP}\left[\mathrm{MO(n)}\right] = \sum_{i=LOMO}^{n} \mathrm{MOOP}\left[\mathrm{MO(i)}\right]$$

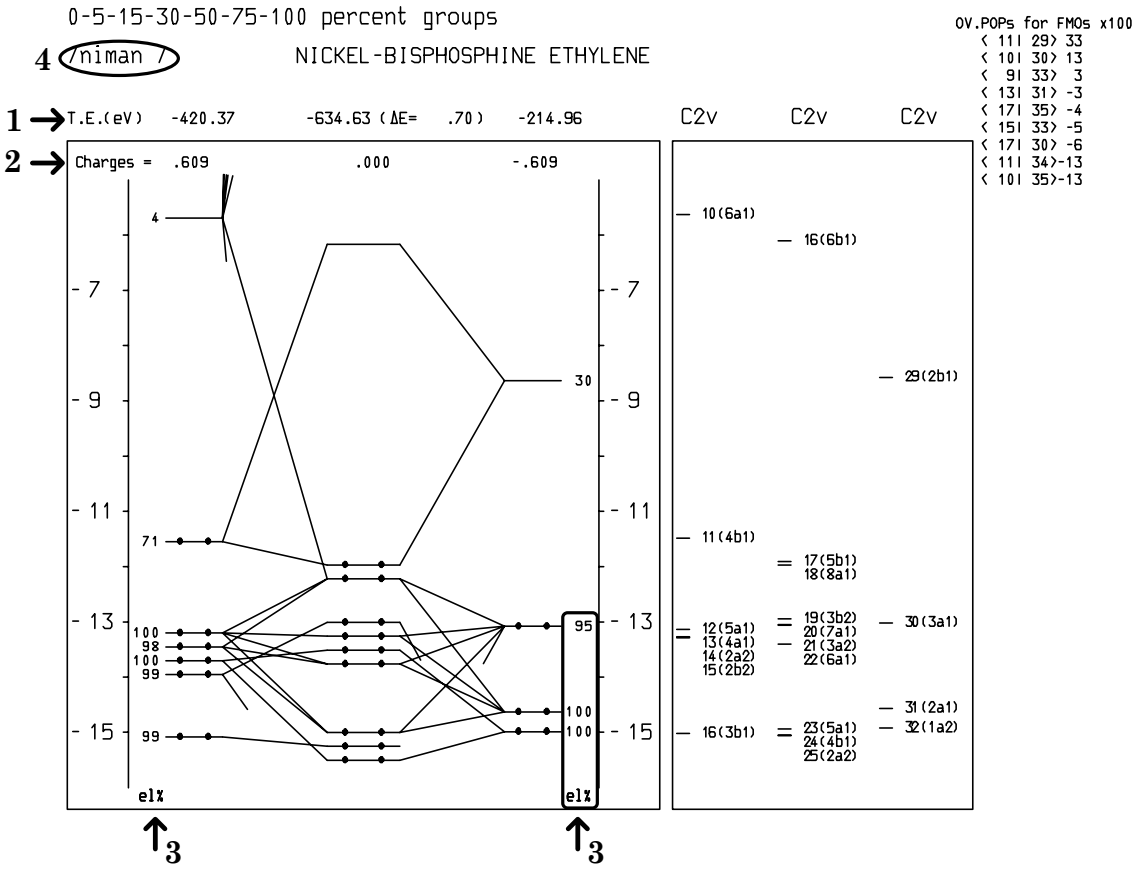
It follows immediately that for n=HOMO, IMOOP(HOMO)= ROP.
Because the IMOOP grows faster than MOOP, the scale of IMOOP is half than for MOOP (but the user can also fix the scale as option, to be able to compare with different plots for different bonds)

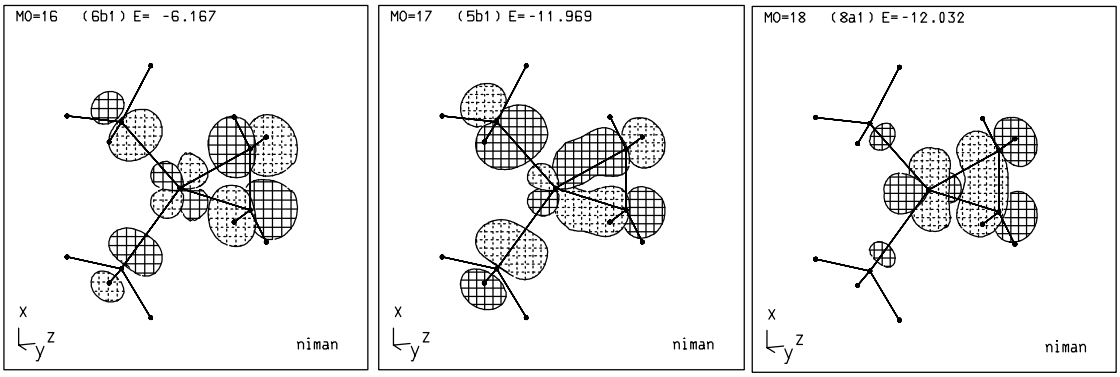
Incidentally, after a FMO calculation, a special MOOP diagram can be constructed which allows to evaluate how much each MO contributes to build up the total overlap population between the two fragments.

For people familiar with solid state calculation, the MOOP diagram is equivalent to the COOP plots with integral.


.Nickel-bisphosphine ethylene
1
C10 - C11
averaged over 1 bonds
2
3
ROP= 1.075
MO range 16 25
integration
-1.20
ROP
1.20
scale for IMOOP
16(6b1)
-7
-9
-11
-13
-15
MOOP
integral of MOOP
1.075
thick for HOMO
17(5b1)
19(3b2)
22(6a1)
(unsplitted)
23(5a1)
(splitted)
scale for MOOP
-0.60
antibonding
"0"
bonding
+0.60
MOs energy


FMO 1
C(10)
Ni(1)
C(11)
FMO 2
X
Z
y





LUMO **HOMO** **HOMO-1**

Description of the option which allows to construct bidimensional surfaces……..
…..

THE STRUCTURE OF THE INPUT FILE in DETAILS

Selected INPUT files are at the end of this document and in the subdirectory FILES. The examination of a few examples will help the user to understand the philosophy for creating the INPUT.
Here follows the description of the INPUT file line-by-line (referred to as directives or cards).
Each directive is characterized by a **keyword** followed by the parameters which can be given in **Free Format** separated either by spaces (or commas). It is **important to follow the order** of the directives as they are given in the summary below. Only, **REM**, **INT**, **INTZM** and **GR** directives can be freely **mixed up**.
The alphanumeric input is not case-sensitive!

DIRECTIVES
(see a summary at the bottom of the file)

Note that those parameters given in curly parentheses **{ }** are optional and may be omitted, while those given in square parentheses **[ ]** are required, even if only equal to zero. If a directive is always required (even though the rest of the line is blank) this is indicated.
A required space is indicated by the character **^**.

TITL ***(always needed)*** {Any 72 alphanumeric characters}

A second Title line is expected if a star '*' appears as the last character of the first TITL line (usually col. 77). Although concisely, use the TITL line to write the contents of the input file. This whole piece of information can be retrieved when making a choice of the dataset in a CACAO session, thus allowing you to remember the details of a particular calculation. This becomes important after the generation of many datasets, since the **memonics associated to a DOS filename** are insufficient (only 8 characters). Notice, in addition, that only the initial 32 characters of the title are reported as the header of the graph reporting either an Interaction or a Walsh Diagrams.

REM ***(optional)*** {Any 72 alphanumeric characters}

This keyword allows to insert a line of remarks in any place of the INPUT file. It is completely ignored by the program.

IMP ***(optional)*** {filetype} {path}
Import coordinates from a given file: filetype can be CSD (cambridge)

PAR ***(optional)*** [cha] {key} {ipoint} {Hückel Constant }

[Default values, if PAR is omitted: Charge=0, key (set for no output of distances), ipoint=1, Hückel constant=1.75]

CHA= Molecular Charge

KEY for distances = The alphanumeric keyword 'd' or 'dist' determines the initial output of the generated cartesian coordinates (from the internal ones) and all of the interatomic separations (Matrix of Distances). If omitted, no distance matrix is generated.

IPOINT= Number of steps to be sequentially calculated in order to produce (in CACAO) a Walsh Diagram. Sets of Free Variables must be supplied in **FV*n*** lines (see below). Each **FV*n*** line contains a set of **IPOINT** values which vary (dependently or independently from the others) at each different step. (If IPOINT is 0 or not specified, IPOINT is set =1).

CON = Hückel Constant with default value = 1.75 *(optional)*

POP [popul] {pop(1) pop(2) ... pop(norb)} ***(optional)***

The directive **POP** allows to fix the electron occupation of the MO levels. If the directive is ***not present***, the MO occupation is ***automatically*** determined, *i.e.* all of the valence electrons in the molecule are accomodated in the appropriate number of MOs which are the lowest in energy. If the energy difference between two adjacent Frontier MOs is <0.1 eV, two or more electrons become unpaired. It may happen, in a Walsh Diagram, that the HOMO and the LUMO become very close in energy only at a certain point. In this case, a change of the ground state is imposed.
If **popul** is ***=0***, the control on the energy differences between the frontier MOs is suppressed (for instance, the HOMO always contains two electrons even if it is practically degenerate with the next empty level).
If **popul** is a ***negative*** integer, its value is taken as the imposed number of the unpaired electrons populating as many different frontier levels. Such an imposed electron distribution remains constant in a Walsh Diagram.
One can fix a particular MO occupation, other than that definable with any of the above strategies. For example, one wishes to have the second LUMO occupied in order to evaluate its effects on the Mulliken Population Analysis. Accordingly, **popul** is set equal to the total number of Molecular Orbitals (=*norb*). This value must be followed by *norb* values, each one defining the number of electrons in any subsequent MO (it can be 0,1 or 2). The *norb* values are supplied by using the adequate number of POP lines. Notice the the order is from the lowest filled (n=1) to the highest empty (n=norb).

CONST ***(optional)*** const1 val1 [const2 val2 ...]

or
const1=expression

It allows to define a series of global constants which can be used in the various **FVn**, **INT**, **INTZM** and **GROUP** directives. The name of a constant must be not longer than 5 alphanumeric characters. No numeric digit is allowed as leading character. Also, some words such as COS, SIN, TAN, FVn, etc. are reserved and cannot be used as constants.
The value of the constant can be either numeric or an expression containing numeric values and/or other constants, already defined**. Expressions must be preeceded by sign** '**=**'. The expression contains typical arithmetic operators or trigonometric functions, such as +,-,*,/,^ or SIN, COS, EXP, etc. The square root is applied as a power with the exponent 0.5 [*e.g*. the square root of 2.0 is 2.0^0.5]. The order of the operations is sequential (namely, there is no jerarchy between the operations +,-,*,/), thus the usage of inclusive parentheses is highly recommended. There may be more than one CONST directives (max. allowed = 90).
The program has some **predefined angular constants**, PI=180, TET=109.4712206, TOH=54.73561, TIH= 37.37736827.
TOH and TIH are useful to define the orientation of the C_3 axis in the Oh and Ih pointgroups, respectively. In fact, TOH is the angle formed by the fourfold and threefold axes in Oh, while TIH is the angle formed by the fivefold and the threefold axes in Ih (see the sample files in the appendix).
Example:
CONST R1 1.4 Ang 119.0 R2 =R1*cos(Ang)

FVn [fv(n,1) fv(n,2) ... fv(n,ipoint)]. ***(optional)***
or
fv(n,1) Step increment
or
expression
Definition of the free variables allow to construct a Walsh Diagram (mono-dimensional array) or prepare data for drawing three-dimensional surfaces (In this case, use other packages such as MatLab, after a reorganization of the data performed interactively with CACAO). For constructing surfaces the free variables are distinguished by plus or minus numbers (e.g. FV1, FV-2, FV3, FV-4, indicate that the variables 1 and 3 are to be varied asynchronously with respect to 2 and 4).
As indicated the syntax for these directives has three different modes. In the first one all of the values aassumed from the variable n up to the final point (ipoint) is specifcally written down. IN the second mode, the user specifies the initial value, the keyword STEP (or S) and the increment. Finally, the user can write an expression depending on constants and on previously defined variables.
Any **FVn** line (max. 20) fixes the actual value assumed by the n^{th} variable at each one of the **IPOINT**s (specified in the PAR directive). FVn can be referenced in any **INT** or **INTZM** or **GROUP** directives, where, symbolically, it represents a geometric magnitude (distance, angle,

torsion, group rotation or translation). As a special case, FVn can appear also in a STO directive, to represent an atomic parameter (orbital coefficient or H_{ii}). In this manner, a special Walsh Diagram can be constructed to follow the effect of the different atomic STO parameters on the MOs.
In the FVn directive, all of the IPOINT values can be explicitly provided or, as an alternative, only the **initial value** is reported, followed by the keyword **Step** (or simply the character **S**) and by the actual **increment**.
Finally, the different values can be calculated at the run time by an explicit **expression** to which the same rules outlined in the directive **CONST** apply (must be preceeded by the sign '='). In this case, the expression depends on previously defined Free Variables (as well as on predefined constants).
In the FVn lines, if n is negative (e.g.: **FV-2**), the corresponding free variable is varied independently from all of the others. This is an useful way of constructing bidimensional surfaces of the order IPOINTxIPOINT (read the section relative to CACAO in order to learn how to exploit this information).

TIP: if one set of free variables remains constant from a certain point on, use the alphabetic character 's' following the first group of values, namely the increment after the last specified value is 0.
E.g. "FV8 110.0 100.0 90.0 s (the angular parameter referenced as Free Variable 8 remains constantly equal to 90.0 from the third up to the last step (=IPOINT).

KEYW [keyword(1) ...]
*(**optional**,* used to control the output of the matrices in EHC calculations).
Choose one or more of the following two-character words in any order:
DI=distances, **WF**=wavefunctions, **OV**=overlap matrix, **OP**=overlap population matrix, **RO**=reduced overlap matrix, **CM**=charge matrix, **RC**=reduced charge matix, **EL**=energy levels, **NC**=net charges, **HM**=Hückel matrix, **DE**=density matrix, **BO**=bond order.

Tips: If no line **KEYW** is present, the output is limited to *Energy levels, Total Energy, Reduced Overlap, Net charges*.
DI is disregarded if the keyword **DIST** is already specified in the **PAR** card)

OVDEL [at.orb(1) at.orb.(2) .....]
*(**optional**)*

The user may set up to nine overlap matrix elements to zero, thereby negating the interaction between these elements. Pairs of negative numbers refer to atomic orbitals, while pairs of positive numbers refer to atoms (so all the overlap integrals between the two selected atoms will be zeroed).
E.g.: OVDEL -23 -34 2 3

The overlap between AO 23 and 34 and the overlaps between all the AO's of atoms 2 and 3 will be zeroed.
To check the selection, it is suggested that the overlap matrix (keyword OV in **KEYW** card) is also printed.

MOOP [MO(1) MO(2) .....]
*(**optional**).*

Used to control the range of Molecular Orbital Reduced Overlap Population matrices to be printed. Only those M.O.O.P. matrices between the pairs given will be printed.
E.g.: The command MOOP 23 34 41 55 causes the M.O..O.P. for levels from 23 to 34 and from 41 to 55 to be printed

SYMM [symbol of the molecule's pointgroup] *(**optional**)*

This directive allows to generate the entire molecule, given the coordinates of the asymmetric part (define only the limited, necessary INT or INTZM directives). If some information is redundant (symmetry related atoms are defined with different INT or INTZM lines), the program realizes this and discard the equal atoms.
The pointgroup Cs indicates mirror symmetry about the plane xy. Specify Csx or Csy for mirror planes coinciding with either xz or yz.
For pointgroups with only σ_v planes (*e.g.* D3h or D5h) impose one of them coinciding with the plane yz. For pointgroups with σ_d planes (*e.g.* D3d) impose one of them coinciding with the plane xz. (Don't forget these constraints when constructing the molecule!
If the great part of the molecule are symmetry related by a given symmetry but a few atoms do not comply with the pointgroup, it is still possible to generating the symmetric part of the molecule by using the SYMM directive. The atoms, for which not all of the given symmetry operations apply, must defined singularly with the opportune INT or INTZM directives. In the latter, it is necessary to append the character '**$**' to the symbol of the atom.

****
skip the following **CELL** *and* **ATOM** *cards if INTERNAL COORDINATES are used*
****

Crystallographic or Cartesian Coordinates

CELL [a, b, c, alpha, beta, gamma] *(**required**)*
Lattice parameters to be used only with crystallographic or cartesian coordinates.
For cartesian coordinates: *a* = *b* = *c*=1, *alpha*=*beta*=*gamma*=90. The latter may not be explicitated (thus a line with the single keyword CELL implies the specification of Cartesian coordinates).

ATOM {nat} [symb] [x, y, z] ***(required)***

SYMB = Atomic Symbol
In general the details specified below for the **ORIG** directive do apply. However in this case, the symbol (of one or two characters) can be followed (without any interrupting blank) by the atom number in order to allow the transfer of the data from any existing crystallographic dataset.
If SYMB=***** , a card defining the STOs (Slater Type Orbitals) for the given atom must follow the whole series of the Internal Coordinates. Otherwise, the predefined STOs in the file PARAM.DAT are used.

X, Y, Z = Cartesian or Crystallographic Coordinates
Tips: if crystallographic coordinates are used, the molecule is translated with the center of mass in the origin.
SYMB may also follow the x, y, z coordinates.

****
If **CELL** *and* **ATOM** *directives have been used, skip the next series of cards.*
****

Internal Coordinates

ORIG [symb] {x, y, z, {ngroup}} ***(required)***

SYMB = Symbol of the atom in the origin. This may be a Real or Dummy atom (=DU).
If SYMB=*****, a card defining the STOs (Slater Type Orbital) for the given atom must follow the whole series of the Internal Coordinates. Otherwise, the predefined STOs in the file PARAM.DAT are used. The usage of SYMB=* supersedes the STOs stored in PARAM.DAT. The symbol '*' applies only to the first atom of the species which is necessary to redefine and for which a STO line is expected. All of the subsequent atoms of the same species must have the same symbol given in the STO line.

X, Y, Z (*optional*) = Cartesian Coordinates of the atom in the origin (if not present, the origin is assumed to be at coordinates 0.,0.,0.)

NGROUP (*optional*) = Number of the rigid Group to which the atom belongs (see **INT** card, for details)

Tip: SYMB may also follow the x, y, z coordinates.

****

WARNING: The following **INT**, **INTZM** and **GR** directives can be given in any order, provided that the needed reference atoms have been defined in previous lines. **INT**, **INTZM** can be alternatively used to define an atom.

INT [symb, at(n1), at(n2), dist, alpha, tau] {ngroup, {tor2}}

Definition of internal coordinates according to a simplified Z-Matrix which implies that three vectors are defined sequentially.

SYMB= symbol of the atom (Real or Dummy) being currently defined.
What stated for the ORIG directive applies in this case too.
Special Symbols are used for Predefined Molecular Fragments. At this moment, the fragments predefined are:
CP=cyclopentadienyl (eta-5), **ME=**methyl, **FO=**phosphine, **AM=**amine, **CM=**carbon monoxide, **BZ=** benzene (eta-6), **PH=** phenyl (eta-1). In all of these cases, the number of the being defined [=AT(N)] refers to the first real atom of the fragment. Be sure to account for the total atoms of the fragment, when defining the atom that immediatly follows the predefined fragment. For example, after the definition of a CP group that follows atom N, the number of atoms is increased by 10, so the next atom is numbered: N+10+1 (see the sample file PSTOOL.IN)
The character '**$**' must be appended to SYMB if the atom in question is to be excluded by the automatic expansion of the molecule, imposed by the presence of the directive **SYMM** (*e.g.*: P$)

AT(N-1) = number of a previously defined atom (A3 in Figure 1) that is taken as the generator of AT(N).

AT(N) = number of the atom being **currently defined** (A4 in Figure 1).

Each atom (with progressive number **N2**) is defined starting from a previously defined atom (**N1**). In the simplified Z-Matrix, the program must refer to the atom from which N1 was generated (say N3) and also to that generating N3 (see Figure 1 and the relative comments). In order to define the first atom departing from that in the origin, the unit vector Z of a right-handed cartesian system must be referred to as the immediately preceeding vector. Moreover, as the second vector, needed for the definition of the definition of the dihedral angle, the unit vector X is used. *(see the Figures at p. 6)*.
The number of the atom is a part of its labelling and is independent of the order with which the atoms are entered in the input file. The number can be optionally negative (allowed range: from -99 to 999) . It may be useful to refer to Dummy Atoms (Symb = DU) with a negative number although it is not necessary. The usege of the Dummies simplify the construction of the Molecule

DIST = Length (in Å) of the vector AT(N-1)--AT(N).

ALPHA = Bond angle defined by the vectors AT(N-2)/AT(N-1) and AT(N-1)/AT(N). If AT(N-2) was never defined, the vector AT(N-2)/AT(N-1) is taken as the cartesian axis Z.

TAU = Torsion angle defined by the vectors AT(N-3)/AT(N-2), AT(N-2)/AT(N-1) and AT(N-1)/AT(N). If AT(N-3) and AT(N-2) were never defined, they are taken as the cartesian axes X and Z, respectively).

NGROUP= *(optional)* Number of the Group to which the atom N, currently being defined, belongs to.
A rigid group is formed by all the atoms for which NGROUP has the same value. The group NGROUPth (see **GR** card) can be rotated and/or translated thus making simpler the construction of Walsh Diagrams for given structural rearrangements. By default all of the atoms belong to rigid group #0, i.e. the molecule is a rigid group itself.

TOR2= *(optional)* Rotation angle of a predefined Fragment about the axis leading to its pivotal atom (e.g., the carbon atom of a methyl group, or the center of a CP ring, eta5-bonded (dummy). If TOR2 is defined also NGROUP must be given (usually 0).

Tips: Any DIST, ALPHA, TAU magnitude can be referenced to the nth Free Variable (**FV*n***). The actual value is that relative to the current step being calculated. If **FV*n*** is preceeded by a negative sign, the sign of corresponding free variable (given in the **Fvn** line) is inverted.
TOR2 cannot be a free variable. In the definition of the special fragments CP or BZ, the DIST value refers to the dummy atom at the center of the ring being defined.
Also, any DIST, ALPHA, TAU magnitude can be defined as an expression (preceeded by the sign '='). However, the expression must contain, in this case, only constants and not free variables.

In general, all the free variables defined vary **synchronously**. However, in order to generate a **bidimensional grid** of steps, one or more free variables can be varied **independently** from the others. This is simply done by referring to the **nth free variable with a negative n**. Thus, if the number os teps is 5 (ipoint =5, in the PAR directive) and five Fv lines have been given, the first 3 free variables can be varied independently from the last 2, by referring to them as FV1, FV2, FV3, FV-4 and FV-5. In this manner, the free variables 4 and 5 assume all of their possible values (five) for any given combination of the first three free variables. Hence, the EHMO calculation is repeated 5x5=25 times.
In this case, the EHC output must be inspectioned directly, as CACAO is not able to process the corresponding binary output and no graphics can be produced (Walsh Diagram or 3D drawings). By manipulating the EHC output file (NAME.OUT) selected magnitudes can be extracted (e.g. Total Energies) and passed to specific applications to generate two-dimensional surfaces.

INTZM [at(n1), symb, at(n2), dist, at(n3), alpha, at(n4), tau] {ngroup, {tor2}}
Definition of internal coordinates of atom at(n1) according to the full Z-Matrix notation.
The required definition of the three vectors implies a reference to three general atoms which have already been defined. Notice that the INT directive implies in which any atom, *e.g.* at(n2), has two fixed generators, [at(n3) and at(n4), respectively]. The structure of this directive is common

to other quantomechanical programs and can be exported to construct their inputs (*e.g.* Gaussian).

AT(N1) = number of the atom being **currently defined** (= A4 in Figure 1)
SYMB = same as in the **INT** lines.
AT(N2) = . number of a previously defined atom (= A3 in Figure 1) that is taken as the generator of AT(N).
DIST = Length (in Å) of the vector AT(N-1)--AT(N).
AT(N3) = . number of any previously defined atom (= A2 in Figure 1) necessary to define the angle alpha (see below).
ALPHA = Bond angle defined by the vectors AT(N3)/AT(N2) and AT(N2)/AT(N1).
AT(N4) = . number of any previously defined atom (A1 in Figure 1) necessary to define the torsion angle tau (see below).
TAU = Torsion angle defined by the three vectors AT(N1)/AT(N2), AT(N2)/AT(N3) and AT(N3)/AT(N4).
NGROUP= same as in the INT directive.
TOR2= same as in the INT directive.

GR [ngroup at1 at2 rotat] {transl} *(**optional**)*
Used for rotating and/or translating rigid molecular fragments about a pivot axis.

NGROUP = The number of the rigid group on which rotation and or translation is operated (refer to **ORIG** and **INT** in order to see how the different atoms can be assigned to a rigid group).

AT1 = One number from the list of the valid atoms defined in the cards ORIG, INT or INTZM cards.
AT2 = The number of a second atom from the same list.
AT1 and AT2 define the vector about which the rigid group must be rotated (clokwise in a right-handed system) or translated (the translation coincides with the direction from AT1 to AT2). For rotations and/or translations about the cartesian axes x, y, z, see the shortcuts below.

ROTAT =Value (deg) of the given angular rotation.
ROT can also be a Free Variable and referred to as FVn. In this case the corresponding series of ramping values must be given in a FVn card. If no rotation but a translation is desidered, ROT must be zero.

TRANSL *(optional)* = Value (Å) for the linear shift of the rigid group. TRANSL can also be a Free Variable. The same criteria for ROTAT do apply in this case, too.

Note: More than one rotational/translational operation can be performed for the same group of atoms, i.e. there may be more directives **GR** for the same group "n". The operations are are carried out sequentially. It is also possible to define a **subgroup** of a larger group. For example, it is possible to translate a whole group and then to rotate only one of its

substituent. This is done by defining the NGROUP variable in directives 10,11 as "n x 10 + m", where "n" is the number of the main group and "m" is that of the subgroup. Two **GR** for n and m must then follow in the order.

If AT1=AT2= (1, 2 or 3) the rigid group is rotated (translated) about the X, Y or Z, principal axes, respectively. The same thing can be done by writing “X”, “Y” or “Z” in place of the pairs AT1, AT2 (namely 1,1 or 2,2 or 3,3).
What already specified in **INT** card for the synchronous or asynchronous variation of free variables holds in this case as well.

STO symb ne ns exps hss [np expp hpp [nd hdd expd1 [c1 expd2 c2]]] *(**optional**)*

*(one STO line is required for any atomic symbol (SYMB) given as * in any of the previous ORIG, INT or INTZM lines)*

The purpose is to redefine the Slater Type Orbitals (STO) and the valence orbital ionization energies (H_{ii}) used in the calculations. Standard parameters are either defined in the program or read from the file PARAM.DAT and, in general, there is no need for a STO directive.
For a given atomic species to be redefined only one STO line is expected. This must be referenced to by an ‘*’ appearing in a line of the type INT, INTZM, ATOM etc. Concerning the sequence of the latter, the ‘*’ must be supplied only for the first atom of the given type, all of the subsequent atoms belonging to the same species being referenced to by the same symbol (SYMB) which appears in the STO line.
The following magnitueds must be given in the order:
SYMB,NE,NS,EXPS,Hss,NP,EXPP,Hpp,ND,Hdd,EXPD1,C1,EXPD2,
C2 , where:
SYMB= atomic symbol - NE= number of valence electrons in the neutral atom - NS, NP, ND = quantum numbers for s, p, d orbitals - EXPS, EXPP = exponents for s and p orbitals - EXPD1, EXPD2, C1 and C2 coefficients for the double expansion of d orbitals - Hss, Hpp, Hdd = Values of the valence orbital ionization energy for s, p, d orbitals.

****
*
The following directives are used only for Fragment Orbital Analysis
****
*

FMO { step} *(**required**)*

The presence of the keyword **FMO** forces the program to perform Fragment Orbital Analysis.

STEP = *(**optional**)* The step number at which the Fragment Orbital Analysis is performed if a multiple set of geometries has been defined by FREE Variables (input file set to create a Walsh Diagram).
***Warning*:** STEP $\neq 0$ supersedes the construction of the Walsh diagram even if IPOINT > 1 in the **PAR** card. Rather a FMO analysis is made only for the molecular model with the geometry fixed by the free variable(s) at

step=STEP (Interaction Diagram). By contrast, even if a **FMO** card is present together with additional next cards, no FMO analysis is performed if STEP=0. In this manner, the required manipulation of the INPUT files in passing from Walsh to Interaction Diagrams and viceversa is minimum.

FKEYW [keywords(1) ...] *(**optional**)*.

These keyword help to select the output of the FMO calculation. Choose one (at least!) or more of the following two-character words in any order: **DI**=distances, **MW**=wavefunctions (AO basis set), **FW**=Fragment wavefunctions (AO basis set), **WF**=wavefunctions (FMO basis set), **OV**=overlap matrix (between atomic orbitals), **TO**=transformed overlap (between FMOs), **OP**=overlap population between FMOs, **CM**=charge matrix, **EL**=energy levels, **RO**=reduced overlap matrix (for fragments), **RC**=Reduced Charge Matrix, **HM**=Hückel Matrix, **EM**=Energy Matrix, **RE**=Reduced energy Matrix, **EP**=Energy Partitioning, **RP**=Reduced Energy Partitioning.

If this FKEYW line is not present, the output is limited to the overlap population and reduced overlap matrices (for fragments). The Fragment Charges and the FMO occupations are ALWAYS typed.

Inportant Tip: the **calculated Molecular Charge**, which is the last piece of information printed by the FMO part of EHC, must be equal to the charge value **CHA** provided by the user in the directive **PAR** card. The calculation is most likely *WRONG* if such a match is not observed. Please recheck the geometry of the molecule in the input file, and, if the problem persists, notify it to the authors including a copy of your input file.

FRAG [nfr, nf(1) … nf(nfr-1) {nc(1)} … {nc(nfr-1)}]
*(**needed for the FMO calculations**)*

NFR = Number of Fragments.
Note : NFR is usually 2 but it can be 3 or more. If there are more than two fragments, the calculations are run properly but the graphic capabilities of CACAO are not available (only the printed output can be examined).

Please notice that only the number of atoms and the charge must be supplied only for the **first NFR-1 fragments**.

NF(n) = Number of atoms in the n^{th} Fragment
NC(n) = Charge of the n^{th} fragment (the values are superseded by those in the **FDEF** cards, if any)

FCOMP [nat(1) nat(2) nat(3) …] *(**optional**)*

Sequentially report in **FCOMP** the original numbers [nat(n)] of the atoms belonging to the first and second (and third, if any) fragment, respectively.

If **nat(n) is negative** all of the atoms up to that number will be assigned to the current fragment. If an **atomic symbol** is inserted in the directive, all of the atoms of the given type will be assigned to the current fragment.
To simplify the input sequence, only the atoms belonging to the first fragment need to be specified and all of the remaining atoms are automatically will be assigned to the second fragment.

Tips:
Use the directive only to change the original numbering scheme of the molecule and to select a particular composition of the fragments. If FCOMP is *not present*, the first **NF1** atoms (see FRAG) from the main original list are assigned to Fragment 1, the subsequent **NF2** atoms to Fragment 2, etc.
If the molecule is constructed by using the directive SYMM (i.e. by exploiting the symmetry of the pointgroup), the automatically assigned numbers of the symmetry generated atoms are unknown at the time when the input files is built. In this case, the user can complete the input file at a later stage. First, he discards any FCOMP directive but he runs EHC with the option mole (>EH . name mol). He checks from the molecular drawings the correctness of the model and collects the proper atomic numbers to be inserted in the FCOMP line after returning to the editor.

An example of the usage of such the directive FCOMP is provided in the INPUT file ETHYLEN.IN (see below).

FDEF [charge {keyword(1) ... }] ***(optional)***

Enter as many directives as the number of fragments NFR. This directives are mainly used to control the numerical output in the MO calculation for the fragments. If only a standard output is required, they may be omitted since it is possible to specify the fragmental charges also in **FRAG** card.
CHARGE = Charge assigned to the fragment [this value supersedes that given in **FRAG** card]
KEYWORDS for selecting the OUTPUT (same as in **KEYW** card).

END

This card is the last instruction. ***(optional)***

INPUT FILE, *selected examples***.**

Copies of the INPUT files illustrated below are found in the directory FILES (subdirectory of MOAN) supplied with the programs.

1) **N2.IN** Molecular nitrogen with FMO analysis. Notice the definition of a Dummy Atom (-1) in the origin (cartesian coordinates 0.,0.,0.). The parameters are separated *either* by spaces *or* commas.

```
TITL  nitrogen molecule. Interaction Diagram
PAR  0 DIST
KEYW EL WF CM OV OP RO NC
ORIG  DU
INT N -1 1  0.564 180 0
INT N -1 2  0.564 0   0
FMO
FRAG 2 1
END
```

2) **O2.IN** Molecular oxygen with FMO analysis for a high spin molecule. The unpairing of two electrons will be automatically determined by the program, in view of the degeneracy of the highest occupied MOs. Notice that all the keywords controlling the output (directives KEYW and FKEYW) have been omitted. Only standard output will be obtained.

```
TITL molecular oxygen. Automatic unpairing of two electrons.
PAR  0 D
ORIG DU
INT  O  -1  1   0.600,180.,0.
INT O  -1  2   0.600,000.,0.
FMO
FRAG   2  1  1 0 0
END
```

3) **ETHYLEN.IN** The molecule of ethylene is built up and its MOs are analyzed in terms of the interactions between the MOs of two methylene fragments. The **FCOMP** card is used to redefine the order of the atoms in the two fragments as the atoms of the first fragment are: C1, H3, H4 and those of the second one: C2, H5, H6. In the FCOMP directive only the atoms of the first fragment could be reported (automatically, the program would reassign the other atoms to the second fragment).

```
TITL ETHYLEN. Analysis of the interactions between two CH2 groups
PAR   0 DIST
SYMM D2h
ORIG DU
INT  -1,1, C  0.70,0.,0.
INT 1,2, H 1.,120.,180.
FMO
FRAG  2 3
END
```

4) **CO2.IN** Bending of a triatomic molecule: CO_2 and costruction of the related Walsh Diagram. The bending angle is a free variable which takes four different values (see Directive 2). The real CO_2 molecule is linear with symmetry D_{ih} ($D_{infinite\ h}$) but the symmetry descends to C_{2v} because of the deformational coordinate. In order to mantain the symmetry constant and create a Walsh Diagram it is necessary to bend slightly the molecule even in the first step. Accordingly, the free variable which applies to the two equivalent angles (defined by the Z-axis and any C-O vector) is equal to 90.1 in the first step.

```
TITL CO2 BENDING
PAR 0 DIST  4
FV1 90.1 s 10
KEYW EL CM NC WF RO
SYMM C2v
ORIG  C
INT 1 2 O 1.150 FV1  0.
END
```

5) **ETHYLDM.IN** Idealized dimerization of two ethylene molecules to generate cyclobutane. **The symmetry forbidness** of the process is highlighted by the Walsh diagram which can be drawn after the EHC calculation. The forbidden interlevel crossing *(Woodward-Hoffmann Rules)* between the HOMO and the LUMO is most evident in the diagram also because the filled and empty MOs are assigned different colours.

```
TITL #Ethyleneʌ Dimerization. Approach of 2 C2H4 and bending of CH2 planes
PAR  0 DIST     5
FV1 1.5 s -0.2
Fv2 0 s 11
KEYW EL CM RO NC DI
SYMM D2h
ORIG DU
INT  -1 -2 DU fv1 180 0.
INT  -2  1 C 0.7,90.,0. ,1
INT   1  2 H 1.01,120.,90.,1
INT   1 -4 DU 1,90,90
GR 1  1 -4  fv2
END
```

The example illustrates the usage of the first Free Variable (FV1) which reduces the separation between two approaching C_2H_4 molecules. The initial distance between the dummy in the origin and that at the C=C midpoint (-2) is 1.5Å (3.0Å between the two molecules). At the last step, the separation between the C_2H_4 molecules is 1.40 Å and all of the four C-C bonds are equivalent. Namely, the newly formed C-C bonds have become equal to the C=C bonds in the two separated ethylene molecules which remain constant all of the times. Although this may seem somewhat unrealistic, it is a good idea with EHMO calculations to mantain bond distances as fixed as possible along a given pathway. The variation of the Reduced Overlap population, when the relative interatomic distance remains fixed, is a good indicator whether the bond itself wants to be elongated or not. This is a valuable chemical information that would be biased and misinterpreted if the bond itself is forced to vary.
This example illustrates another useful feature, namely the Group rotation (and/or translation) by using a **GR** card. Along the pathway the four CH_2 planes reorient themselves relative to the original C=C bonds (up to 45°). Actually, any group is rotated about an axis passing through the carbon atom and perpendicular to the C_4 plane (the dummy atom -4 is defined to the purpose). The free variable FV2 determines step by step the amount of pinning back of the four CH_2 planes.
As soon as the EHC program starts (option MOL) the steps along the pathway can be easily examined (+ and – buttons).

6) **PSTOOL.IN** Piano Stool Complex.
This example illustrate the usage of PREDEFINED groups (4 carbon monoxide CM and 1 cyclopentadienyl CP). Also, the possibility of changing the atoms parameters with respect to the standard ones (contained in the File PARAM.DAT) is pointed out. In this case, the atomic symbol Mo is replaced by a '*' in the corresponding Internal Coordinate Definition line (INT). This means that a **STO** line is expected.

```
TITL piano stool
PAR 1 DIST
KEYW EL CM RO NC
ORIG *  0.  0. 0.
INT CM 1 2 2.0 70. 0.
INT CM 1 4 2.0 70. 90.
INT CM 1 6 2.0 70. 180.
INT CM 1 8 2.0 70. 270.
INT Cp 1 10 1.8 180. 0. 0 90.
STO MO 6 5 1.96 -8.34 5 1.92 -5.24 4 4.54 -10.5 .6097 1.9 .6097
END
```

7) **NIOLEF.IN** Nickel bis(phosphine) ethylene.
A study of the rotation of ethylene to check the presence of an energy barrier. The example again exploits the rotation of a rigid body or GROUP. Notice the definitions of the phosphines as predefined groups (a single line).

```
TITL  Nickel-bisphosphine ethylene
PAR   0 DIST  4
FV1  0.1 30. 60. 89.9
KEYW  EL CM NC RO
ORIG  NI 0. 0. 0.
INT fo 1 2  2.2 90. 130.
INT fo  1 6  2.2 90. 230.
INT DU 1 -1  1.7 90. 0.
INT C -1 10   .7 90. 270.      1
INT C -1 11   .7 90. 90.       1
INT H 10 12    1.01 120. 90.   1
INT H 10 13    1.01 120. 270.  1
INT H 11 14    1.01 120. 90.   1
INT H 11 15    1.01 120. 270.  1
GR   1  1 -1  fv1 0.
END
```

9) **NAPHT.IN** 1,8-naphthyridine.
The example shows the usage of crystallographic coordinates. The latter are taken from: Dapporto, Ghilardi, Mealli, Orlandini, Pacinotti, *Acta Cryst.* C40, **1984**, 891. As a useful exercise go into the Molecular Editor (select command E from the menu) and optimize the symmetry (Command O). The molecule is idealized to C_{2v} pointgroup.

```
TITL 1,8- NAPHTHYRIDINE
PAR  0 d
KEYW EL WF RO NC
CELL  6.135,10.407,11.258,90.,117.76,90
ATOM  N .3228,.4512,.2064
ATOM N .2755,.2680,.0806
ATOM C .2250,.5416,.2479
ATOM C -.0265,.5471,.2164
ATOM C -.1826,.4553,.1354
ATOM C -.2363,.2626,-.0075
ATOM C -.1284,.1745,-.0524
ATOM C .1288,.1810,-.0040
ATOM C -.0889,.3584,.0836
ATOM C .1680,.3586,.1237
ATOM H .3313,.6065,.3024
ATOM H -.0863,.6157,.2533
ATOM H -.3581,.4547,.1143
ATOM H -.4137,.2603,-.0334
ATOM H -.2165,.1117,-.1149
ATOM H .2060,.1212,-.0322
END
```

10) **O2H2.IN** The following example shows a cis-trans isomerization of O_2H_2 that occurs *via* a torsion of the two OH groups about the O-O vector. This is achieved by forcing the free variable fv1 to assume opposite values (fv1 and -fv1).

```
TITL Hydrogen Peroxide  CIS-TRANS isomerization
PAR    0 DIST  4
FV1  0.1 s 30
KEYW EL CM OP RO NC
ORIG  DU
INT  O -1  1  0.740 180. 0.
INT  O -1  2  0.740 000. 0.
INT H  1  3  1.05 110. fv1
INT H  2  4 1.05 110. -fv1
END
```

11) **FERROCEN.IN** To obtain the staggered conformation for ferrocene we use the torsion of the one of the internal defined group CP by 36°.

```
TITL Ferrocene
PAR  0 DIST
KEYW EL WF CM RO NC RC
ORIG  FE 0. 0. 0.
INT Cp 1 2  1.8 180 0
INT Cp 1 12  1.8 0 0. 0 36.
FMO
FKEYW CM NC RO TO WF
FRAG 2 1 2 -2
END
```

12) **C60PT1.IN** The file contains the input for calculating the MOs of the adduct between C_{60} and the metal fragment $Pt(PH_3)_2$. A fragment orbital analysis analysis is also performed. The overall symmetry is C_{2v} but that of C_{60} is I_h. Since the program does not recognize the latter, it is convenient to construct the C_{60} within the D_{2h} pointgroup. The metal atom and the phosphines bound to it have a "$" sign following their symbols so that the D_{2h} molecular expansion does not apply to them. Notice that for FMO analysis the first fragment is made up with 60 atoms namely only the carbon ones (FCOMP line). It is interesting in the CACAO session to draw a MOOP diagram relative to the overlap population between fragments MO by MO.

```
TITLE C60Pt1
KEYW EL NC
SYMM D2h
ORIG Du
INT   90  91 DU 3.2594,148.2752,0.
INT   91  11  C 1.19091,90.,36.,1
INT   91  12  C 1.19091,90.,108.,1
INT   91  13  C 1.19091,90.,180.,1
INT   91  14  C 1.19091,90.,252.,1
INT   91  15  C 1.19091,90.,324.,1
INT   90  92 DU 3.2574,90.,-31.7248
INT   92  16  C 1.19091,90.,90.,3
INT   92  17  C 1.19091,90.,162.,3
INT   92  18  C 1.19091,90.,234.,3
INT   92  19  C 1.19091,90.,306.,3
INT   92  20  C 1.19091,90.,18.,3
INT   90  93 DU 3.2584,58.2752,90.
INT   93  21  C 1.19091,90.,36.,5
INT   93  22  C 1.19091,90.,108.,5
INT   93  23  C 1.19091,90.,180.,5
INT   93  24  C 1.19091,90.,252.,5
INT   93  25  C 1.19091,90.,324.,5
INT   90   1 PT$ 5.4300,180.,0.
INT    1   2 FO$  2.25,128.8,0.
```

```
INT    1   3 FO$  2.25,128.8,180.
FMO
FKEYW RO EL OP
FRAG  2 60
FCOMP   C
END
```

13) **CAMBR.IN** This File shows how to import a structure from the Cambridge Structural Database. The File in question is fosf2.dat (also supplied in the directory Files) and contains all of the structures of trimetallic clusters with three phosphido bridges. You can select one of them interactively by its refcode (if the structure consists of cations, anions, etc., only the opportune residue may be selected). Then the model can be simplified and the symmetry optimized within the molecular editor. Upon the SAVE option, the file CAMBR.IN will contain the cartesian coordinates. This input file can be opportunely manipulated to make Interaction (FMO) or Walsh Diagrams (select atoms as groups which can be rotated or translated with the directive GR).

```
TITL cambridge
IMP CSD C:\MOAN\Files\FOSF2.DAT
END
```

14) **NMEI2SU.IN** This file shows how to prepare the data for a bidimensional surface (7x7) to be used with plotting programs such as MATLAB. The case in question is relative to the adduct between amine and iodine and the independent elongation of the N-I and I-I bonds is performed (see the usage of FVn and FV-n, free variables)

```
TITL  N(CH3)3-I2  PES for the independent elongation of N-I and I-I distances
PAR   0 Dist 7
fv1 2.1 s .2
fv2 2.55 s .2
SYMM C3v
ORIG I
INT I   1  2 fv-2 180.  0.
INT N  1  3   fv1 0 0
INT Me  3  4 1.50 110 0
END
```

15) **OCTA.IN**. This example shows how to construct an Octahedral molecule, formed by a cube of Nickel atoms with the faces capped by carbon atoms. It is important in order to have a correct assignement of the MO symmetries that one fourfold axis (containing one C atom) coincides with the Z axis. The other two fourfold axes must coincide with the bisectors of the XY axes. Notice that the threefold axis containing the Ni atoms is defined with the internal angular constant TOH (see directive CONST).

```
TITL
SYMM Oh
```

```
ORIG DU
INT 1 2 Ni 1.7 TOH 0
INT 1 3 C 2.3 0 0
FMO
FRAG 2 8
FCOMP NI
END
```

Summary of the Directives

TITL	Title card	First compulsory line
REM	comment line	optional, used at any place
IMP	Import Coordinates from given file	optional
	First group of directives (in any order)	
PAR	General parameter card	optional
METH	Computational Method	optional
POP	Mode for populating the MOs	optional
CONST	Definition of geometrical constants	optional
FV*n*	Values of the Free Variable *n* at the different steps (can also be an expression)	optional
KEYW	Keyword card to control the EHC output	optional
OVDEL	Overlap deletion card	optional
MOOP	Range of M.O.O.P. matrices printed	optional
SYMM	Pointgroup Symbol	optional
	Second group (crystallographic or cartesian coordinates)	
CELL	Unit cell dimensions card	required, as first
ATOM	Atomic coordinates card	required, one per atom
	Second group (internal coordinates)	
ORIG	Atom in the origin	required, as first
INT	Internal coordinates definition with simplified Z-Matrix	required (may alternate with **INTZM**)
INTZM	Internal coordinates definition with complete Z-Matrix	required (may alternate with **INT**)
	Third group (optional)	
GR	Rigid group rotation or translation line	optional
STO	Atomic parameters redefinition line	optional (referred to in **ATOM**, **ORIG**, **INT** or **INTZM** lines)
	Fourth group (optional, follow the order)	
FMO	Request for a FMO calculation	required, as first
FKEYW	Fragment MO keyword card	optional
FRAG	Fragment definition card	required
FCOMP	Fragment composition card	optional
FDEF	Fragment information card	optional
END	End of calculation	optional